LA-UR 95-2132

Approved for public release; distribution is unlimited

Traffic Jam Dynamics in Stochastic Cellular Automata

Authors: K. Nagel, M. Schreckenberg

Traffic Jam Dynamics in Stochastic Cellular Automata

Kai Nagel, Los Alamos National Laboratory and Santa Fe Institute, U.S.A. and Michael Schreckenberg, Universität Duisburg, Germany

95ATS089

Simple models for particles hopping on a grid (cellular automata) are used to simulate (single lane) traffic flow. Despite their simplicity, these models are astonishingly realistic in reproducing start-stop-waves and realistic fundamental diagrams. One can use these models to investigate traffic phenomena near maximum flow. A so-called phase transition at average maximum flow is visible in the life-times of jams. The resulting dynamic picture is consistent with recent fluid-dynamical results by Kühne/Kerner/Konhäuser, and with Treiterer's hysteresis description. This places CA models between car-following models and fluid-dynamical models for traffic flow. — CA models are tested in projects in Los Alamos (USA) and in NRW (Germany) for large scale microsimulations of network traffic.

1 Introduction

Simulation obeys a basic trade-off between resolution, fidelity, and scale. Many questions in transportation simulation currently can only be treated using high-resolution models (i.e. resolving each individual traveller) and looking at large spatial and temporal scales (e.g. for land use forcasts). In that situation, it is straightforward to think about reducing fidelity as much as possible. With respect to vehicular traffic flow, this means to find a *minimal* model of driving behavior without loosing features which are believed to be essential for *large scale* questions. One solution to this challenge are cellular automata [25, 27] models of traffic [5, 4, 23].

These models serve a multitude of purposes: They run fast on computers, which allows both quick testing of different approaches as well as extensive statistical analysis, and they are simple enough for analytical treatment. All this helps answering the question of a *minimal fidelity* (minimal complexity [3]) model of driving behavior. In addition, the high computing speed also offers a potential for real time applications.

Schreckenberg and Nagel [22] give a more physics based view on why such models are useful.

2 The Stochastic Traffic Cellular Automaton (STCA)

The basic computational model is defined on a one-dimensional array of L sites with open or periodic boundary conditions. This could, for example, be a link in a road network. Each site is either occupied by one vehicle, or empty. Each vehicle has an integer velocity with values between zero and v_{max}, where we often use $v_{max} = 5$ for reasons stated below. The number of empty sites in front of a vehicle is denoted by *gap*. For an arbitrary configuration, one update of the system consists of the following four consecutive steps, which are performed simultaneously for all vehicles:[1]

1.) **Acceleration:** If the velocity v of a vehicle is lower than v_{max} and if there is enough space ahead, then the speed is increased by one: *If* ($v < v_{max}$ *&* $v < gap$) *THEN* $v := v + 1$.

[1] Note that *either* rule 1 *or* rule 2 applies, but never both.

```
.................3......4.........3...000.00...3.........3....2..3....4.......
....5...............4.......5........0000.01......4.........3...3...3.....4...
.........4..............5........2...0001.0.2.........4........3...2...4......
..4..........5...............4.....0.000.01...3...........5.......1..3.....5..
5.....5...........4..............1.0.000.1.2.....4.............3...1....3.....
.....4.....4..........4...........00.001..1..2.......5............1.2......3..
.........4.....5..........5.......00.01.1..2...3..........5........1..3.......
.5...........4......4..........1..00.1.1.1...2....4............3....1....4....
......4..........5......5.......0.00..1.1.1....3......4...........2..1.......4
4.........5...........4......1..1.00...0.1.1......3.......5.........0.2.......
....5..........4..........3...2..000...0..0.1........3.........3....1...3.....
.........4.........4.........2..0001...0..1..2..........4.........1..2.....3..
.............4.........5.......0001.1..1...2...2............4......1...2......
.................5..........1..001.0.1..1....3...3..............3...1....3....
..5...................4......0.00.01..1..1......2...3..............0.1......3.
.......4..................1..1.01.1.1..2..1.......3....3...........1..2.......
.3.........5...............2..00.1.1.1...0.1.........4....4.........1...2.....
....4...........5............000..1.1.2..1..1............4....5......1....2...
```

Figure 1: Simulated traffic at a density of 0.09 cars per site. Each new line shows the traffic lane after one further complete velocity-update and just before the car motion. Empty sites are represented by a dot, sites which are occupied by a car are represented by the integer number of its velocity. Cars are moving from left to right. — One clearly sees the jam wave moving opposite to the traffic direction.

2.) **Slowing down (due to other cars):** Else if the next vehicle ahead is too close, speed is reduced to *gap*: ... *ELSE IF* $(v > gap)$ *THEN* $v := gap$.

3.) **Randomization** (*which is applied after rules 1 & 2*): With probability p, the velocity of each vehicle (if greater than zero) is decreased by one.

4.) **Car motion:** Each vehicle is advanced v sites.

Note that, because of integer arithmetic, expressions like for example $v < gap$ and $v \le gap - 1$ are equivalent.

Since this model involves discrete space, discrete time, a small number of discrete states per cell, and a local and completely synchroneous update, this model is formally a (stochastic) cellular automaton.

Already this simple model shows nontrivial and realistic behavior, for example start-stop-waves which are visible in space-time-plots (Fig. 1), or qualitatively realistic fundamental diagrams (Fig. 2). That reinforces our claim that already fairly simple models of driving dynamics can be rather realistic on more macroscopic scales.

Step 3 is essential in simulating realistic traffic flow since otherwise the dynamics is completely deterministic. When following through the rules for special cases like a single car on an empty road or a car approaching a thick traffic jam, one finds that it condenses three different behavioral patterns into one computational rule: (i) fluctuations at maximum speed; (ii) retarded acceleration; (iii) over-reactions at braking. Without this randomness, every initial configuration of vehicles and corresponding velocities reaches very quickly a stationary pattern which is shifted backwards, i.e. opposite to the vehicle motion, one site per time step [11, 12, 15].

The model can also be seen as a discrete particle hopping model [9]. Simpler versions of the CA model can be related to the Lighthill-Whitham theory of traffic flow. See [22] for more details.

3 Traffic jam dynamics

The statistics of the traffic jams in the STCA can be analyzed in a systematic way. Roughly speaking, a so-called phase transition takes place at the density ρ_c corresponding to maximum *average* flow; at higher densities than that traffic jams *can* survive for infinitely long times. This phase transition can be described using concepts from the physics of critical phenomena.

More precisely, assume a situation with laminar traffic flow at a given homogeneous density ρ_o and with a homogeneous flow $q_o = v_{max} \cdot \rho_o$ (cf. Fig. 3). Now assume a small disturbance

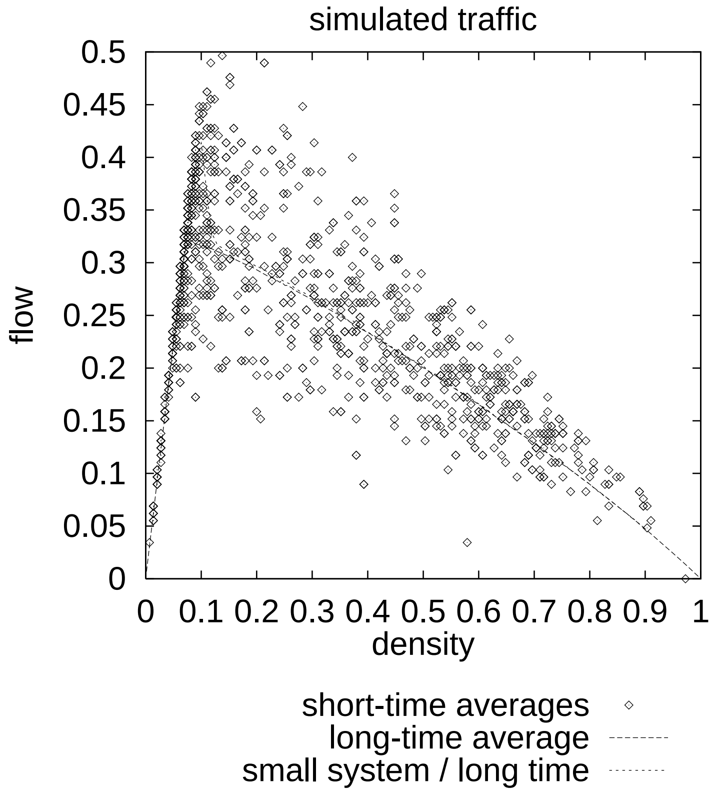

Figure 2: Fundamental diagram of the model (throughput versus density). Points: Averages over short times (100 iterations) in a sufficiently large system ($L = 10,000$). Broken line: Long time averages (10^6 iterations) in a large system ($L = 10,000$). Dotted line: Long time averages in a small system ($L = 100$).

```
.........5..........5...........5...........5...........5..........5..........
..5...........5..........5...........5...........5...........5..........5.....
.......5...........5..........5...........5...........5...........5..........5
5...........5...........5..........5...........5...........5...........5......
.....5...........5...........5..........5...........0...........5...........5.
..........5...........5...........5..........5......1................5........
...5...........5...........5...........5..........2..1....................5...
........5...........5...........5...........5.......0.1.......................
.5...........5...........5...........5...........1..1..1......................
......5...........5...........5...........5.......2..2..2.....................
5..........5...........5...........5...........3....2..1..2...................
.....5..........5...........5...........5.........2...1.2...2.................
..........5..........5...........5...........5......2..1..3...3...............
...5...........5..........5...........5...........3...1.2....3...3............
........5...........5..........5...........5.........0.2..2.....2...3.........
.5...........5...........5..........5...........4....0...2..2.....3....3......
......5...........5...........5..........5..........00.....1..2......4....3...
...........5...........5...........5..........5.....01......1...3........2...3
....5...........5...........5...........5..........00.2......2.....3.......2..
.........5...........5...........5...........5.....00...2......3......3......3
...5..........5...........5...........5...........000.....2.......3......4....
........5..........5...........5...........5......000.......2........3.......5
.5...........5..........5...........5...........0.001.........3.........4.....
......5...........5..........5...........5......0.01.2...........3..........5.
...........5...........5..........5...........1.1.0.1..3............4.........
```

Figure 3: Space-time plot of traffic at density $\rho = 0.85$ in the so-called cruise control limit, i.e. fluctuations at maximum speed are artificially set to zero. At $t = 5$, we emulate a disturbance by setting the velocity of a randomly selected vehicle to zero. One clearly sees the growing disturbance.

of this laminar flow, for example by setting the velocity of one randomly selected car to zero. Many different choices for the local perturbation give rise to the same large scale behavior. The perturbed car eventually re-accelerates to maximum velocity. In the meantime, though, a following car may have come too close to the disturbed car and has to slow down. This initiates a chain reaction – an emergent traffic jam.

Now, *in the average*, one finds three different behaviors, depending on ρ_o:

- For relatively low densities, the emergent traffic jam quickly dissolves.
- For relatively high densities, the emergent traffic jam grows forever, that is the inflow into the jam $q_{in} = q_o$ is higher than the outflow q_{out}, and the average number of cars in a jam, $\langle n(t) \rangle_{all}$, behaves as

$$\langle n(t, \rho_o) \rangle_{all} \sim t \cdot (q_{in}(\rho_o) - q_{out}) \ ,$$

  where $\sim$ means "proportional to for $t \to \infty$" and $\langle \ldots \rangle_{all}$ means the ensemble average over all initiated jams, i.e. the average over many simulations using ρ_o but different Monte Carlo seeds.
- Somewhere in between, that is when ρ_o is such that $q_{in} = q_{out}$, the emergent jam *in the average* neither grows or shrinks. This density is called the critical density $\rho_c = q_{out}/v_{max}$. Note that the critical point is given by the outflow behavior, which is in turn given by the acceleration behavior of drivers.

See [16] for more details.

It has to be stressed again that this description is only true in the ensemble average, i.e. for each given ρ_o we averaged over many different Monte Carlo realizations. For an *individual* jam, it happens rather often that even above ρ_c the jam eventually dies out; others however keep on growing forever and therefore compensate in the average.

This *average* picture is phenomenologically consistent with the fluid-dynamical model of Kühne/Kerner/Konhäuser [8, 7, 20, 24] (KKK model), except that one has to replace $\langle n(t) \rangle_{all}$ by the amplitude $A(t)$: In the fluid-dynamical model, below a certain density ρ_c, $A(t)$ *de*creases exponentially; above ρ_c, one finds, at least initially, exponential growth, and exactly at ρ_c, the amplitude neither grows nor shrinks. See [18].

The CA picture is also consistent with some newer mathematical car following models [2], and also with Treiterer's observation of a hysteresis effect: Laminar flow above the critical density ρ_c is possible for some time, but eventually traffic breaks down and separates into (i) jams, and (ii) laminar flow *at* ρ_c in between jams [18].

However, the STCA goes beyond the fluid-dynamical and (mathematical) car following models mentioned above as it allows for a stochastic evolution of the traffic jam. This is expected to be extremely important in traffic networks near capacity, where spill-back from disturbances may grow back into upstream intersections, which may cause a complete network break-down. Here, a deterministic model either always predicts network breakdown or always not, whereas a stochastic model allows to predict the *probability* of such a break-down event.

It is interesting to note that a theoretic argument for the jam dynamics suggests that traffic jam dynamics is "universal", which is a well defined concept in the theory of critical phenomena and means that probably many and maybe all one-dimensional traffic models with "reasonable" rules display, again in the thermodynamic limit and in a certain coarse-grained description, the same traffic jam dynamics. Depending on the needs for the traffic question under consideration, the simplest version of this jam dynamics may well be good enough and more complicated models then only cost resources.

4 Computational speeds

We have performed extensive tests of computational speeds of different implementations of the CA on different computer architectures [14, 19, 21, 15]. Especially for parallel machines, system size becomes a crucial quantity since, for example, a certain system size may run efficiently on a small number of processors but inefficiently on a large number of processors. For simplification, the following numbers therefore are extrapolations from a simulated system of size 10 000 km (single lane), with a CA density of $\rho = 0.1$, i.e. 133 333 vehicles; for more details see the above references. Some conclusions from the tests can be summarized as follows:

- CA models are in principle amenable to single bit coding, thereby stuffing all 32 (or 64) bits of a computer word with information and processing this information simultaneously using word-wise logical operations. These approaches run extremely fast on traditional vector-computers like the Crays. We reached real time limits of $4 \cdot 10^6$ km or $53 \cdot 10^6$ veh sec/sec (vehicle seconds per second) on a NEC SX-3.
- On non-vectorizing parallel machines though, single bit coding is not much faster, and it is also much less flexible for more ambitious uses where one would, say, want to keep track of travelers' plans. We still reached, for example, a real time limit of 260 000 km or $3.5 \cdot 10^6$ veh sec/sec on an 32-node Intel iPSC hypercube, or 220 000 km ($2.9 \cdot 10^6$ veh sec/sec) on a 32-node CM-5 (not using the vector units).
- An implementation of the freeway network of NRW (including freeway intersections and lane changing) slowed the simulation speed down by less than a factor of 2. We reached 290 000 km or $3.8 \cdot 10^6$ veh sec/sec on a 64-node Intel Paragon.
- Extrapolations for a large but existing machine, the 1024-node CM-5, let us expect $1.7 \cdot 10^6$ km or $22 \cdot 10^6$ veh sec/sec, i.e. enough simulation speed for the whole Los Angeles area in real time.

5 Simulations of the freeway network of NRW

The model has been used to make microscopic simulations of the freeway network of the German land NRW [19, 15]. Here we will describe some simulations which have been undertaken to test the usefulness of a CA approach for simulating route choice behavior in realistic network configurations.

In the simulations, there are many travelers with different origin-destination pairs. Travelers have route plans (paths) so that they know on which intersections they have to make turns in order to reach their destinations. Each traveler has a choice between 10 different paths. Each traveler chooses a path, the microsimulation is executed according to the plans of each traveler (no re-planning during the trip), and each traveler remembers the performance of his/her option.

Each traveler tries each option once. Afterwards, she usually chooses the option which performed best in the past, except that, with a small probability p_{other}, another option is chosen randomly, in order to update the information about other options.

This approach – giving each agent a set of options and let each agent act on the basis of the performances of these options – is a simplified version of Holland's classifier systems [6, 1].

The simulations were based on a digital version of the freeway network of NRW (see [19, 15] for details). The code is written for parallel computers using message passing, in principle for an arbitrary numbers of computational nodes (CPNs). In practice, two Sparc10 workstations, coupled via optical link and using PVM 3.2, were used. That gives the idea that experiments such as the following are already possible with a still modest amount of hardware, and that the consistent use of parallel supercomputers will allow systematic analysis of much larger systems.

Apart from the network and the individual trip plans, the simulation is kept as simple and straightforward as possible. This includes single directional lanes (i.e. only one lane in each direction) and over-simplified ramps [13, 15].

The specific simulation set-up is described in the following. The simulation is an approximation of long distance traffic through NRW. Let us denote by *boundary points* the points where the freeways cross the borders of NRW. In the way the network was prepared [19, 15], these are the only open ends of the network. At each of these endpoints, 2000 vehicles were assumed to be in an ordered queue with the desire to enter as early as possible. Each vehicle had a destination, which was one of the other boundary points of the networks, and which remained the same through all repetitions of the simulation. The destinations were chosen randomly, with a higher probability of choosing destinations which were far away; using different distributions gave only small differences on the level of detail which is discussed here [15].

Then, each vehicle gets a list of 10 different paths to reach its destination. These lists have been pre-calculated for all allowed O-D-pairs, and contain the 10 geometrically shortest paths which do not use the same node twice ([10], see [15]). Each vehicle individually now decides which path to use. In the first day (= period), each vehicle uses the shortest path; during the subsequent 9 days, each vehicle randomly selects one of the not yet tested options. Afterwards, it selects, as mentioned above, the option with the best performance, apart from the probability $p_{other} = 5\%$ to re-test one of the other options, which is then chosen randomly.

After these preparations, the microsimulation starts. Vehicles are updated according to the standard dynamical rules, and they change segments when they are at an intersection. Each vehicle then follows its plan until it reaches its destination, where it notes the arrival time t_{arriv}, i.e. the current iteration step of the simulation, which is used as performance criterion for this specific path. When this is the first time this specific path has been used, this is used as the first guess; for repeated trials of the same route, both a myopic and an averaging scheme were tested without visible differences.

After all vehicles have reached their destinations and recorded the above information, the simulation is restarted, where all vehicles have the same initial position and the same destination as before, but may choose a different path according to the rules described above.

Figs. 4 demonstrates the result of the learning algorithm. Both the top and the bottom figure uses exactly the same initial configuration of cars with their individual destination. Both figures are snapshots of the situation after 6000 iterations (100 minutes). The top figure shows the situation when every driver follows the geometrically shortest path. Meanwhile, in the bottom figure, drivers act according to their previous experiences, i.e. they usually use the path where they were fastest in the past. Note that, generally speaking, people "learn" according to the programmed rules to equilibrate the jams between different paths, so that all options are equally slow.

Further experimentation reveals that the day-to-day pattern of these simulations after the learning phase is fairly stable. Together with the result that also neither the learning algorithm (myopic vs. averaging) nor the specific distance distribution do not matter very much, one can conclude quite in general that many of the results are robust, in spite of the stochasticity of the model.

Both the TRANSIMS project in Los Alamos [26] and the "Verkehrsverbund NRW" in Germany plan to use CA approaches as a microsimulation option.

6 Summary

Already a simple, grid-based particle hopping model yields realistic traffic jam behavior and realistic fundamental diagrams. The model was used to investigate the regime of maximum flow. It was found that there is a phase transition at a critical density ρ_c; above ρ_c traffic jams in the average never dissolve, therefore defining average maximum flow after the break-down of

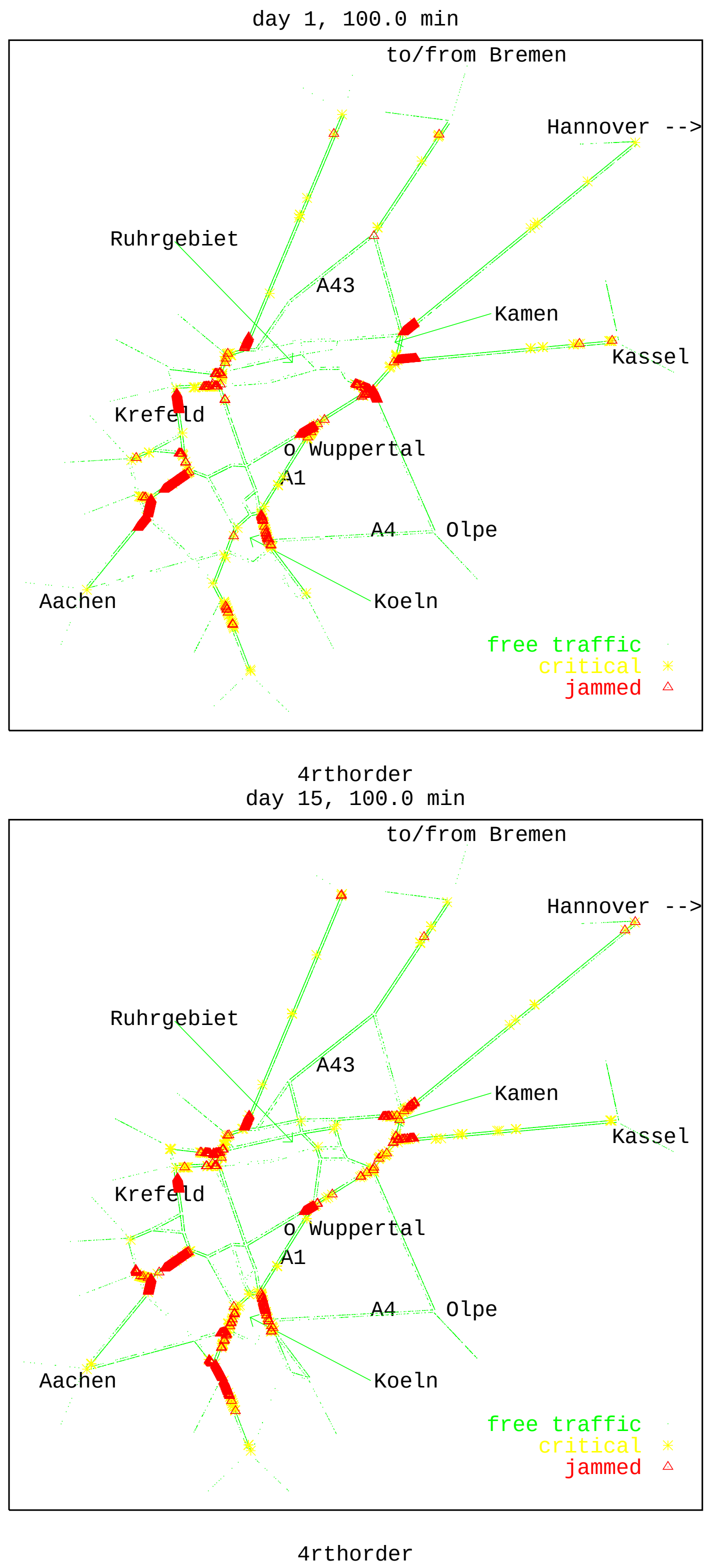

Figure 4: *Top:* Situation at the "first day" after 6000 iterations (100 minutes), when trips through the network are chosen with a fourth order preference for long trips, and when all drivers follow the geometrically shortest path. — *Bottom:* Situation at "day 15" after 6000 iterations (100 minutes), for the same initial conditions as for the top figure, but where drivers have "learned".

laminar flow. This behavior is similar to recent fluid-dynamical and continuous car-following theories of traffic flow, as well as with Treiterer's hysteresis description. Implementations on different supercomputers confirm that the computational speed is sufficient to perform regional traffic simulations in faster than real time. Results from simulations of the freeway network of the German land North-Rhine-Westfalia show in a more practical case, that this model makes simulations based on individual behavior possible with only a modest amount of hardware.

References

[1] B. Arthur, Inductive reasoning, bounded rationality, and the bar problem, American Economic Review (Papers and Proceedings) **84** (1994).

[2] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, Structure stability of congestion in traffic dynamics, Japan Journal of Industrial and Applied Mathematics **11**(2), 203 (1994) ; M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama, Dynamical model of traffic congestion and numerical simulation, Phys. Rev. E **51**(N2), 1035 (1995).

[3] C. Barrett (personal communication).

[4] M. Cremer, J. Ludwig, A fast simulation model for traffic flow on the basis of Boolean operations, Mathematics and Computers in Simulation **28**, 297 (1986).

[5] D. L. Gerlough, Simulation of freeway traffic by an electronic computer, in *Proc. 35th Annual Meeting*, Highway Research Board, Washington, D.C., 1956.

[6] J. H. Holland, Using classifier systems to study adaptive nonlinear networks, in *Lectures in the sciences of complexity*, edited by D. Stein (Addison-Wesley Longman, 1989), p. 463.

[7] B. S. Kerner and P. Konhäuser, Cluster effect in initially homogenous traffic flow, Phys. Rev. E **48**(4), R2335 (1993) ; B. S. Kerner and P. Konhäuser, Structure and parameters of clusters in traffic flow, Phys. Rev. E **50(1)** 54 (1994).

[8] R. Kühne, Verkehrsablauf auf Fernstraßen, Physikalische Blätter **47(3)**, 201 (1991) ; R. Kühne, Traffic patterns in unstable traffic flow on freeways, in *Highway Capacity and Level of Service*, edited by U. Brannolte, (Balkema, Rotterdam, 1991); R. Kühne, Freeway Speed Distribution and Acceleration Noise – Calculations from a stochastic continuum theory and comparison with measurements, in *Proc. Int. Symp. on Transpn. and Traffic Theory* (MIT press, Cambridge, 1987); R. Kühne, Macroscopic freeway model for dense traffic stop-start waves and incident detection, *9th Int. Symp. Transpn. Traffic Theory* (VNU Science Press, 1984).

[9] T. M. Liggett, *Interacting particle systems* (Springer, New York, 1985).

[10] C. Moll (personal communication).

[11] K. Nagel and M. Schreckenberg, A cellular automaton model for freeway traffic, J. Phys. I France **2**, 2221 (1992).

[12] K. Nagel and H.J. Herrmann, Deterministic models for traffic jams, Physica A **199**, 254 (1993).

[13] K. Nagel and S. Rasmussen, Traffic at the edge of chaos, in *Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, edited by R. A. Brooks and P. Maes (MIT Press, Cambridge, MA, 1994), p. 222.

[14] K. Nagel and A. Schleicher, Microscopic traffic modeling on parallel high performance computers, Parallel Computing **20**, 125 (1994).

[15] K. Nagel, *High-speed microsimulations of traffic flow*, Ph.D. thesis, University of Cologne, 1995.

[16] K. Nagel and M. Paczuski, Emergent Traffic Jams, Phys. Rev. E **51**, 2909 (1995).

[17] K. Nagel, M. Schreckenberg, Traffic jam dynamics in stochastic cellular automata, 95ATS089, this issue.

[18] K. Nagel, Particle hopping models and traffic flow theory (in preparation).

[19] M. Rickert, *Simulationen zweispurigen Autobahnverkehrs mit Zellularautomaten*, Master Thesis, University of Cologne, 1994.

[20] M. Rödiger, Chaotische Lösungen nichtlinearer Wellengleichungen mit Anwendungen in der Verkehrsflußtheorie, Master Thesis, University of Münster, 1990.

[21] M. Schreckenberg, A. Schadschneider, K. Nagel, and N. Ito, Discrete stochastic models for traffic flow, Phys. Rev. E **51**, 2939 (1995).

[22] M. Schreckenberg and K. Nagel, Physical modelling of traffic with stochastic cellular automata, 95ATS094, this issue.

[23] H. Schütt, Entwicklung und Erprobung eines sehr schnellen, bitorientierten Verkehrssimulationssystems für Straßennetze, Schriftenreihe der AG Automatisierungstechnik TU Hamburg-Harburg No. 6, 1991 (unpublished).

[24] B. Sick, *Dynamische Effekte bei nichtlinearen Wellengleichungen mit Anwendungen in der Verkehrsflußtheorie*, Master Thesis, University of Ulm, 1989.

[25] D. Stauffer, Computer simulations of cellular automata, J. Phys. A **24**, 909 (1991).

[26] TRANSIMS—The **TR**ansportation **AN**alysis and **SIM**ulation **S**ystem project, TSA-DO/SA, Los Alamos National Laboratory, U.S.A.

[27] S. Wolfram, *Theory and Applications of Cellular Automata*, (World Scientific, Singapore, 1986).